Introduction

Java has been a part of developers' vocabularies since 1995. At first it was thought of as being a nice, neat little language that could do some amazing things for the Internet. However, the language soon matured, and it still kept its simple approach. Developers started to realize the awesome power of a clean uncluttered alternative to C/C++.

It wasn't long before visionaries in the industry discovered that Java could be further extended into an "enterprise" language. Thus J2EE (Java 2 Enterprise Edition) was born. This has also matured into a solid base for running three-tier, web-based, enterprise systems.

If anyone doubts the industrial strength of these systems, there are now a wealth of blue-chip corporations using J2EE. They use IBM WebSphere and other enterprise systems to create very large, robust, and "externalized" systems.

The dot-com boom may have adjusted itself somewhat, but it is by no means gone. The statement that the Gartner group made a few years ago, that corporations would have to externalize their data or lose out to competitors that have, is still very valid. Can you imagine working with a bank that did not offer online banking? They wouldn't survive for very long if their competitors were all "webified"!

So, in 2001, one of the most innovative ERP companies, SAP, saw an opportunity to bring Java into its development environment. SAP has said that Java and ABAP will coexist as development languages. With Web Application Server (WAS) 6.40, we have seen this become a reality. Although there is still room for improvement (isn't there always?) we now have a credible SAP platform for delivering web services.

Make no mistake—SAP is very serious about Java. It is not a passing fancy or an attempt to be fashionable. When I first lectured about Java to ABAP programmers in Europe in late 2002, SAP already had 35 internal projects using and developing Java. SAP has developed a "flavor" of J2EE to fit inside WAS.

In this Foundations book, we will be looking at the standard J2EE and the new Java EE 5. You will find it easy to use the SAP-specific APIs once you have mastered the standard version. Rest assured, though, that I will explain everything from an ABAP programmer's point of view. I will also show you the NetWeaver way where appropriate.

As I write this, Sun has recently renamed Java (Standard Edition) 1.5 to *Java 5*. Sun is also releasing Java 5 Enterprise Edition (Java EE 5), and this has been done as part of the Java Community process. This is important, because SAP (and others) have been part of this process.

WAS 6.40 does not currently use Java EE 5, but considering that technologies like Enterprise JavaBeans (EJB) 3.0 make life easier for developers, it's a certainty that SAP will include it soon. Rather than covering the old way of doing things, we will explore the latest technology so that you will be adequately armed for the next release.

Many books have leapt into discussions of how SAP employs Java *without* adequately explaining the basics. This book aims to reverse that trend by leading the reader through *bite-sized* lessons with *simple* examples that stress the points in the lessons.

Clearly, in my opinion, Java is a lot of fun. If you need an illustration of this, check out the Robocode project at `http://robocode.sourceforge.net/`.

I hope you enjoy this book. Remember to have fun with Java!